СВИТЪК С УРОЦИ ОТ ТЕМАТА "КОМПЮТЪРНО МОДЕЛИРАНЕ"





КОМПЮТЪРНО МОДЕЛИРАНЕ И ИНФОРМАЦИОННИ ТЕХНОЛОГИИ



Ангел Ангелов • Даниела Дурева-Тупарова • Георги Тупаров Катерина Марчева • Катя Михова-Стоянова

СВИТЪК С УРОЦИ ОТ ТЕМАТА "компютърно моделиране" за **5. Клас**

Знаци, използвани в уроците

– задача за включване в портфолиото на ученика

🔁 – задача с по-голяма трудност

Ресурсните файлове към уроците са организирани в папки, съдържащи в името си номера на съответния урок.



компютърно моделиране

Създаване на графични изображения с изучаван език за блоково програмиране

Да припомним!

Растерното изображение се състои от **пиксели**. При завъртане, накланяне или промяна на размерите му то **губи** част от качеството си.

Векторното изображение се представя с геометрични фигури, описани с математически формули. При промяната му се запазва качеството на изображението.

Графични изображения се създават с помощта на **графични редактори**.



Как да създадете нов герой?

Важно!

Ако от векторен режим преминете в растерен режим, изображението ще си остане растерно, дори и да се върнете отново във векторен режим. Задача 1. Използвайте векторния режим на графичния редактор на средата за визуално програмиране и създайте герой – балон за летене. Поставете центъра на героя в средата на "дъното" на коша. Създайте два костюма на героя, като промените цветовете на балона и на коша.

В частта за герои първо изберете бутона за избор на герой (спрайт), а след това – инструмента **Нарисувай**. Отваря се графичният редактор за създаване на костюми в режим на векторна графика.

Спрайт	Спрайт	1		↔ x	0	1 у	*
Показване	0	ø	Размер		Нарисув	хай	1
							Q

С бутона Преобразуване в растерен режим може да преминете към растерна графика. Когато сте в режим на растерна графика, може да преминете към режим на векторна графика с бутона





Програмата **Paint** е графичен редактор за растерна графика. С нея може да създавате нови герои и сцени, които да използвате във вашите проекти по компютърно моделиране. Може да използвате и **вградения в средата за визуално блоково програмиране графичен редактор**. С него може да се създават **растерни** и **векторни** изображения. Векторният режим позволява по-лесно да се редактират отделни елементи (обекти) от изображението.

Цвят за запълване или очертаване може да изберете с инструментите Запълване 🔜 или Подчертаване 🗔 .

При избор на инструмента за запълване се появява панелът вдясно. С плъзгачите от него може да изберете цвят, наситеност и яркост, а с инструмента пипета 🧪 – произволен цвят от платното за рисуване. Другите инструменти, които може да използвате, са:



Задача 2. Използвайте растерния режим на графичния редактор и създайте герой – хвърчило. Разгледайте и експериментирайте с различните инструменти в избрания режим. Създайте два костюма на хвърчилото, като промените цветовете му.

Задача З. Създайте код за балона от задача 1 така, че той да:

a) започне да се движи от позиция x: -100 и у: -50 и да изчертае квадрат със страна 150 стъпки със зелен цвят на молива и дебелина 4; 6) смени костюма си и да изчертае правоъгълник със страни 200 стъпки и 100 стъпки, като използва син цвят на молива и дебелина 6.



Задача 4. Създайте код за хвърчилото от задача 2 така, че то да започне да се движи от центъра на сцената и с върха си да начертае: а) успоредник със страни 40 и 30 стъпки и остър ъгъл между тях 45°. Променете дължините на страните и ъгъла така, че героят да изчертае квадрат, ромб и правоъгълник.

6) успоредник с върхове в точките с позиции (0; 0), (100; 0), (140; 60) и (40; 60).

Въпроси и задачи

1. Създали сте герой във векторен режим и сте преминали в растерен режим. Може ли отново да преобразувате героя във векторно изображение?

2. В графичния редактор на средата създайте различни герои, като използвате геометрични фигури.





Забележка



Упътване



3. Във векторния режим на графичния редактор създайте герой – хвърчило. Напишете код, с който героят да начертае фигурите, образувани от равностранни триъгълници, като започне от центъра на сцената.



Създаване и използване на собствени блокове или подпрограми

Да припомним!

За придвижване на героите по сцената в средата за визуално блоково програмиране се използват блокове за движение, определяне на посоката на движение и завъртане, които са групирани в менюто .

урок 2

Повтарящи се действия се кодират с блокове за цикъл. За проверка на условие се използват блокове за разклонение. Блоковете за цикъл и проверка на условие се намират в менюто . За създаване на анимация на героите се използват блокове за смяна на костюми от менюто .





Какви са разликите в кодовете на пингвините?



Какво е предназначението на собствените блокове, или подпрограми?

Задача 1. Отворете файла penguins_1.

a) Стартирайте кода и наблюдавайте движенията на героите. Какви действия извършват те?

6) Разгледайте кода на големия пингвин – героя *Penguin*. Колко и кои блокове са използвани за преместването му до крайната цел?
6) Разгледайте кода на малкия пингвин – героя *Penguin_kid*. Колко и кои блокове са използвани за преместването му до крайната цел?
г) Кои са повтарящите се блокове в кодовете на двата пингвина?

Задача 2. Отворете файла penguins_2.

a) Стартирайте кода и наблюдавайте движенията на героите. Какви действия извършват те? Има ли разлика в действията, които извършват двата пингвина в тази задача, и действията им в задача 1?
6) Разгледайте кода на големия пингвин. Колко блока са използвани за преместването му до крайната цел? Какви нови блокове откриваш?

6) Разгледайте кода на малкия пингвин. Колко блока са използвани за преместването му до крайната цел? Какви нови блокове откриваш?

г) Кои са повтарящите се блокове в кодовете на двамата герои?

В действията на пингвините в двете задачи няма разлика. Разлика има в кодовете им. Придвижването на големия пингвин в задача 1 е осъществено с два блока за цикъл с 10 повторения на преместване и смяна на костюм. В кода на малкия пингвин също е използван блок за цикъл с 10 повторения на преместване и смяна на костюм. Блоковете в задача 2 са по-малко. Между тях има нови блокове в розов цвят.

В изучаваната среда за визуално блоково програмиране може да създавате **собствени блокове** (наричат се още **подпрограми**), които да включват блокове от менютата или други вече създадени от вас блокове. Собствените блокове се използват, когато в кода има **многократно повтарящи се части** с цел намаляване на размера на кода и по-лесното му управление. За да създадете собствен блок:

Моите

 Изберете менюто (Му Blocks).
 В списъка с блокове се появява бутон Създаване на блок (Make a block).
 Изберете го.

3. Отваря се прозорецът Създаване на

блок, в който има няколко полета. На мястото на име на блок въведете името на вашия блок и потвърдете с бутона Добре.

4. В полето за писане на код на средата се появява розов блок дефинирай (define). Добавете необходимите блокове към него.



🔚 Код

Пвижение

🖌 Костюми

Съзлаване на блок

Моите Блокове

1

Собствените блокове действат само за героя или за сцената, за които са създадени. За да използвате подпрограма в кода на друг герой, трябва да прехвърлите с влачене кода, описан в блока **дефинирай**, върху другия герой и да промените името на блока.

- За да редактирате дефиницията на вече създаден блок:
- **1.** щракнете с десния бутон на мишката върху блока дефинирай;
- 2. изберете Редактиране;

3. в отворилия се прозорец за създаване на собствен блок направете съответните промени.

За да изтриете дефиницията на собствен блок:

1. премахнете блока от кода;

2. щракнете с десния бутон на мишката върху дефиницията му и изберете Изтрий Блок.

Kak ga създадете собствен блок?

Забележка

Препоръчително е името на собствения блок да отговаря на действията, които ще извършва героят с него, или на промените, които ще се случват на сцената.

Важно!

С опцията *Стартиране без* обновяване на екрана (Run without screen refresh) действията на героя се изпълняват, като на екрана се показва само крайният резултат.

Може ли кодът на собствен блок да се използва от няколко героя?

Kak ga pegakmupame u изтривате дефиницията на собствен блок?

Речник

Define – Дефинирай Make block – Създай блок My Blocks – Моите блокове Run without screen refresh – Стартиране без обновяване на екрана

Въпроси и задачи

1. За какво се използват собствените блокове (подпрограми)?

2. Ани създала код за придвижване на балон и хвърчило със смяна на техните костюми на всеки 20 стъпки. За движението на балона използвала собствен блок. Ще действа ли този блок и за хвърчилото? Защо?

3. Отворете файла figures_1. Стартирайте кода. Проследете движението на червената точка. Редактирайте дефиницията на блока **Чертай квадрат със страна 40 стъпки**, като изберете опцията *Стартиране без* обновяване на екрана. Как се визуализира квадратът? 4. Отворете файла **penguins_1**. Променете кода на големия пингвин, като създадете собствен блок за придвижването му. Използвайте блока, за да придвижите малкия пингвин по стъпките на големия пингвин.

5. Отворете файла **balloon_1**. Създайте блок за придвижване на балона нагоре с 15 стъпки и смяна на костюмите му. Създайте блок за задаване на начална позиция на балона долу в центъра на сцената. Балонът да се движи постоянно и когато докосне ръб на сцената, да се връща в началната си позиция.

Създаване и използване на собствени блокове или подпрограми

(упражнение)

Да припомним!

В изучаваната среда за визуално блоково програмиране може да създавате собствени герои в два режима на вградения графичен редактор – векторен и растерен.

урок 💆

Може да създавате **собствени блокове**, често наричани **подпрограми**. При създаването на такъв блок се задава име на блока и се дефинират действията в него. Собственият блок може да се използва многократно в кода на героя, за който е създаден.

Въпроси и задачи





a) в растерен режим, когато се разтегля изображението, качеството му не се променя;

6) във векторен режим може да се променят отделни части от изображението, без това да се отрази на качеството му?

2. Вярно ли е, че:

a) когато в кода има многократно повтарящи се части, за да се намали размерът на кода и той да се управлява по-лесно, се използват собствени блокове (подпрограми);

6) един и същ собствен блок може да се използва директно от всички герои на сцената?

3. Отворете файла **balloon_kite**. Създайте блокове за придвижване

на балона нагоре с 15 стъпки и смяна на костюмите му, а на хвърчилото – нагоре с 20 стъпки и смяна на костюмите му. Създайте блок за задаване на начална позиция на героите в долния край на сцената. Когато даден герой докосне ръба, да се връща в началната си позиция. Героите да се движат постоянно.

4. Отворете файла **houses**. Разгледайте собствените блокове в кода. Използвайте ги и довършете кода така, че моливът да изчертава къща като дадената вляво.







5. Преработете кода от задача 4, като създадете собствен блок **Чертай къща**, с който да изчертаете 6 къщи така, както е показано на фигурата.



6. Отворете файла **flowers**. Преработете кода за изчертаване на цветята, като дефинирате собствени блокове. Променете кода така, че след изчертаване на петото цвете да се изчертаят още 6 цветя, които да се разположат на случайно място на сцената.



7. В нов проект вмъкнете квадратна мрежа със страна на квадратите в нея 30 стъпки. От библиотеката с герои вмъкнете героя *Pencil (Молив)* и намалете размера му на 25%. Поставете центъра на молива във върха му. Създайте собствени блокове за изчертаване на лодка, подобна на дадена-



та на фигурата. Създайте код за придвижване на лодката надясно.

Упътване

Лодката на фигурата се състои от корито с формата на трапец и триъгълно платно. Може да начертаете трапеца, като използвате равностранни триъгълници (виж задача 3 в края на урок 1). "Основата" на лодката на фигурата е 60 стъпки, а на "палубата" – 120 стъпки. Отсечките, които свързват "основата" и "палубата" на лодката, са 60 стъпки, ъгълът между "палубата" и отсечката, свързваща я с основата, е 60°. Определете ъглите на завъртане. Стрелките показват посоките на движение. Червената точка е началната позиция на молива.

Упътване

Преценете кога трябва да бъде пуснат и вдигнат моливът.



Упътване

Размерът на герой се задава в проценти. За да намалите размера на героя, използвайте





Създаване и използване на собствени блокове или подпрограми с входни данни

Да припомним!

Собствените блокове (подпрограми) се използват, когато в кода има многократно повтарящи се части. Може да включват блокове от менютата на средата или други вече създадени собствени блокове. Името на собствения блок трябва да подсказва предназначението му. Може да редактирате дефиницията на собствен блок, като щракнете с десния бутон на мишката върху блока дефинирай, изберете Редактиране и нанесете корекциите в появилия се прозорец. За да изтриете дефиниция на собствен блок, първо трябва да премахнете блока от кода и след това да изтриете дефиницията му.



Как да създавате и използвате собствен блок с входни числови или текстови данни?



Задача 1. Отворете файла figures_2. Стартирайте кода. Разгледайте дефинициите на блоковете Чертай квадрат със страна 40, Чертай квадрат със страна 60 и Чертай квадрат със страна 80. По какво си приличат и по какво се различават?

В основната част на кода (фиг. 1) е направено обръщение към три различни собствени блока. В дефинициите и на трите блока е използван един и същ код за изчертаване на квадрат.



Разликата е в числото, посочено в блока премести се с 10 стъпки

В средата за визуално програмиране може да създадете собствен блок (подпрограма). Като входни данни при неговото извикване може да използвате **различни стойности**.

За да създадете блок с входни числови или текстови данни, използвайте Добави поле за въвеждане число или текст от прозореца за създаване на блок. Може да добавите и етикет, който да подсказва предназначението на полето за въвеждане. За да създадете блок за изчертаване на квадрат със страна, чиято дължина се подава като **входни данни** за блока:

1. Изберете менюто **Моите блокове** и от него – **Създаване на** блок.

2. В прозореца Създаване на блок задайте име на блока.

3. Използвайте Добави поле за въвеждане число или текст (Add input number or text) – променлива, чиято стойност ще се използва в блока, и Добавяне на етикет (Add a label) (фиг. 2). За да бъде кодът по-лесно четим, променете текста число или текст например на Дължина (име на променливата). Потвърдете с бутона Добре.

Важно!

Етикетът подсказва предназначението на входните данни и е много полезен, когато задавате повече входни данни за блока.



4. В дефиницията на блока ще се появи полето **Дължина**.

5. С мишката влачете полето Дължина до мястото в кода, на което искате да се използва неговата стойност. В случая това е блокът



Кодът на преработената задача 1 може да се види на фиг. 3.

Като входни данни на собствен блок (подпрограма) може да задавате стойности на променливи или изрази. Например в кода от фиг. 4 за изчертаването на фигурата от фиг. 5 е добавена променливата **стъпка**. В тялото на цикъл с 5 повторения се извиква блокът **Чертай квадрат със страна** с входни данни аритметичен израз, който съдържа променливата **стъпка**: **стъпка** + 20.











Как да създавате и използвате собствен блок с входни логически стойности?



Знаете ли, че?

Логическите стойности и операторите не, и, или се наричат още *булеви (Boolean)*. Те носят името на *Джордж Бул* – създател на булевата алгебра, която е в основата на съвременните компютри.

Речник

Add a label – Добави етикет Add input – Добави поле за въвеждане Add input boolean – Добави поле за въвеждане на булев израз

Задача 2.

a) Отворете файла **test1**. Това е игра тест с въпроси по компютърно моделиране. Отговаря се с Да или Не, като правилен отговор може да бъде както Да, така и Не. Изиграйте играта. Разгледайте кода на бухала. Колко пъти се прави проверка дали отговорът е верен или не? Кои блокове са използвани за проверката?

6) Отворете файла **test2**. Това е същата игра от а). Разгледайте кода. Колко пъти се прави проверка дали отговорът е верен или не? Кои блокове са използвани?

В кода от файла **test2** е използван собствен блок за проверка на условие, което има **логическа стойност** истина или лъжа (True или False).

Стъпките за **създаване на блок с входни логически стойности** са подобни на стъпките за създаване на блок с входни числови или текстови данни.

1. Изберете менюто **Моите блокове** и от него – **Създаване на блок**.

2. В прозореца Създаване на блок задайте име на блока.

3. Изберете Добави поле за въвеждане да-не въвеждане (Add

input boolean) (фиг. 6). За да се ориентирате по-лесно, на мястото на boolean може да запишете текст на български език, например Правилно. В дефиницията на блока ще се появи полето Правилно.



Фиг. 6

4. С мишката влачете полето **Правил**но до мястото, където искате да се използва стойността му.

В задача 2 това е условният блок. Част от кода е дадена на фиг. 7.

дефинираи	Провери отговор Правилн
ако Пра	авилно тогава
кажи Пр	равилно! за 2 сек
промени	точки 👻 с 🔳
иначе	
F	

Фиг. 7

Въпроси и задачи

1. Дани създал код със собствен блок за изчертаване на равностранен триъгълник. Какво ще се случи, ако при създаването на блока е активирал опцията *Стартиране без обновяване на екрана*?

2. Отворете файла **test1**. Променете кода, като използвате собствен блок за проверка на отговора. **3.** Отворете файла **task**. Създайте код за изчертаване на фигурата, като използвате собствен блок за изчертаване на равностранен триъгълник.



Създаване и използване на собствени блокове или подпрограми с входни данни

(упражнение)

VPOK

Δa припомним!

В средата за блоково програмиране може да се създават собствени блокове, като в тях се до-

бавят променливи, чрез които да се въвеждат числа, текст или логически стойности. Може да се добавят етикети, които подсказват предназначението на входните данни.

Въпроси и задачи

1. Ани дефинирала дадения блок за изчертаване на фигура и го използвала в код.



Каква фигура ще се изчертае при използването на блока



с въведените входни данни ?

2. Дани създал собствен блок към герой – риба, на сцената, и го използвал по следния начин:





пълнението на кода?

З. Отворете файла task3.

а) Експериментирайте с различни правилни многоъгълници и дължини на страните им. Каква фигура се получава при увеличаването на броя на страните на многоъгълника?

6) Създайте собствен блок за изчертаване на правилен многоъгълник по зададени брой и дължина на страните му. Използвайте блока, за да преработите кода на молива.

В) Създайте блок за изчертаване на успоредник. Използвайте го, за да начертаете фигурите вдясно.

4. Отворете файла task4. Декорите на сцените съдържат въпроси по компютърно моделиране и информационни технологии. Създайте собствен блок за проверка на отговорите и го използвайте, за да създадете игра викторина. Добавете променлива, с която да броите верните отговори.



Упътване



Създаване на образователен проект със средствата на изучаван език за блоково програмиране – алгоритми за размяна на стойности и броене на елементи

Да припомним!

За осъществяване на диалог между героите

урок б

или **синхро**низиране на действията им в средата за визуално блоково програмиране се използват блоковете:

когато получа		
разпространи	съобщение1	
разпространи	съобщение1	

Препоръчително е името на съобщение да дава информация за действията или героя, който го изпраща.

Променливата е част от паметта на компютъра,

в която се съхраняват данни – текст, числа, логически стойности. Всяка променлива има **име** и стойност. Тя може да се използва многократно в кода, да участва в различни оператори и команди, да променя стойността си. Променливи се създават от менюто и избор на бутона Създаване на Променлива . Блоковете за работа с променливи са:

направи	моята променлива 🔻 на 🚺
промени	моята променлива 👻 с 🚺
покажи пр	ооменлива моята променлива 👻
скрий про	менлива моята променлива 🔻

При създаването на променлива се определя дали да

се отнася само за даден герой, или за всички герои.

Ведната има 80 мл оранжада, а в другата Околтия ими. Сконстия ми.

Как може да размените стойностите на две променливи?



Задача 1. Стартирайте файла **Change_juice_1**. Проследете как се разменя съдържанието на двете чаши. Рагледайте целия проект. **а)** Колко са героите? Колко костюма имат героите *Glass lemonade*,

Glass orange и Glass empty?

б) Проверете колко и кои променливи са създадени.

6) Колко и кои съобщения са използвани за синхронизиране на действията на героите и диалога между тях?

г) Разгледайте кода на героя *Glass empty*.

В кода от задача 1 е приложен един от алгоритмите за размяна на стойностите на две променливи – алгоритъм за смяна на стойностите на две променливи с използване на допълнителна променлива. Този алгоритъм се прилага като част от много други задачи и алгоритми. С някои от тях ще се запознаете в следващия урок. Нека са дадени две променливи с имена *A* и *B*, чиито стойности ще се разменят, и променлива *C*, която е допълнителна (помощна).

1. На променливата *C* дай стойността на *A*.

2. На променливата A дай стойността на B.

3. На променливата *В* дай стойността на *С*. Вляво е дадена схема на изпълнение на алгоритъма.

Описаният алгоритъм е използван в кода на на героя *Glass empty* за размяна на стойностите на двете променливи **Чаша1** и **Чаша2**. Блоковете, с които е направена размяната, са:



Задача 2. Стартирайте файла fraction_number. Разгледай кола.

a) Кои блокове са използвани за създаването на собствения блок

Провери вида на дроб?

6) Какви променливи са създадени и кои блокове са използвани за промяна на стойностите им?

в) Какво е предназначението на използваните променливи?

За да **преброите** точките в дадена игра, животните от определен вид, правилните дроби или каквито и да е други предмети или елементи в код, трябва да използвате променливи, в които да се съхранява текущият брой на елементите.

На променливите, които използвате за броене, трябва да зададете начални стойности с блока **направи**. За да увеличите или да намалите стойността на дадена променлива, използвайте блока **промени**. За начална стойност и за стойност, с която да се увеличава или да се намалява стойността на променливата за броене, може да използвате и аритметични оператори или изрази.

Забележка

Променливите в koga се nokaзват в уголемен размер. За да промените начина на nokaзване на една променлива, щракнете с десния бутон на мишката върху нея и изберете начин на пokaзване.



Kak ga преброите различни елементи?



Въпроси и задачи

1. Колко и кои са стъпките в алгоритъма за размяна на стойностите на две променливи с помощта на допълнителна променлива?

2. Опишете как се броят елементи в код.

3. Отворете файла **Change_juice_2**. В кода на героя *Glass empty* са добавени блокове, с които се "изпразват" съответните чаши. Променете кодовете на героите така, че:

a) за начални стойности на променливите да се използват текстове: за **Чаша1** – "Лимонада", за **Чаша2** – "Оранжада", за **Чаша3** – "Празна";



6) в блоковете за "изпразване" на чашите да се запише текст "Празна".

6) Тествайте резултата. Какво се случва със стойностите на променливите, които са показани на сцената? **4.** Отворете файла **reciprocal**.

a) С кои блокове е направена проверка дали въведените числител и знаменател на обикновената дроб са различни от нула?

6) Колко променливи са създадени, кои са те и какви са началните им стойности?

5. Създайте код, с който да се въвеждат числител и знаменател, различни от нула, на 10 обикновени дроби, да се преброят правилните дроби и на сцената да се показват дробта и реципрочното ѝ число.



Важно!

Не забравяйте да правите проверка на въвежданите от клавиатурата данни! Създаване на образователен проект със средствата на изучаван език за блоково програмиране – намиране на минимален/ максимален от три елемента и подреждане на три елемента по големина

Да припомним!

Стойностите на две променливи А и В може да се разменят по алгоритъм, в който се използва до-

урок 7

- пълнителна променлива С:
- **1.** На *С* дай стойността на *А*. **2.** На *А* дай стойността на *В*.
- **З.** На *В* дай стойността на *С*.

направи	размяна 🔻	на	число1
направи	число1 🝷		число2
направи	число2 👻		размяна

За броене на елементи в код в среда за блоково програмиране се използват променливи и блокове за промяна на стойностите им.



Как да намерите найголямото от три числа с размяна?

Знаете ли, че?





Задача 1. Отворете файла max_number. Стартирайте кода. Разгледайте го.

- а) Колко променливи са създадени? Кои са те?
- 6) Кои блокове са използвани за въвеждане на трите числа?

6) Кои блокове са използвани за размяна на стойностите на две променливи?

В задача 1 е приложен алгоритъм за намиране на най-голямото измежду три числа, в който последователно се сравняват две съседни числа и се разменят стойностите им.

Нека са дадени променливите А, В и С, чиито стойности са числа. Алгоритъмът е следният:

1. Сравняват се стойностите на A и B. Ако A > B, се разменят стойностите на А и В (В вече ще съдържа по-голяма стойност от А).

2. Сравняват се стойностите на *B* и *C*. Ако B > C, се разменят стойностите на В и С (С вече ще съдържа най-голямата стойност).



ако число1 > число2 тогава
направи размяна 🕶 на число1
направи число1 - на число2
направи число2 - на размяна
ако число2 > число3) тогава
направи размяна 🔻 на число2
направи число2 🕶 на число3
направи число3 - на размяна

Задача 2. Отворете файла sort. Стартирайте го.

a) Тествайте кода с различни стойности за постиженията на роботите.

6) Опишете действията на героите на сцената.

6) Разгледайте кода на всеки герой. Почетната стълбичка също е герой с име *podium*. Колко променливи са създадени? Обяснете кои променливи са използвани за всеки герой.



г) Разгледайте кода на героя *podium* и открийте частите от него, които се описват с вече изучени алгоритми.

g) Разгледайте кодовете на героите *R1*, *R2* и *R3*. По какво си приличат и по какво се различават?

В кода на задача 2 са създадени 7 променливи:

• 3 за точките на роботите (по 1 за всеки робот – точкиR1, точкиR2 и точкиR3) – началните им стойности се въвеждат от клавиатурата;

• 3 за всеки робот – **число1**, **число2** и **число3**, като първоначалните им стойности са равни на броя на въведените от клавиатурата точки на всеки робот;

• една помощна променлива за размяна на стойностите на две променливи размяна.

В кода на героя *podium* е използван алгоритъмът за размяна на стойностите на две променливи от урок 6.

За класирането на героите е приложен методът на "мехурчето" (bubble sort). Той се състои в следното:

Нека са дадени три променливи *A*, *B* и *C*, чиито стойности са числа.

1. Намира се най-голямото от трите числа (с алгоритъма от задача 1).

2. Сравняват се стойностите на другите две променливи и се разменят.

3. Стойностите на променливите се подреждат: *A* < *B* < *C*.



Начало

След като се изпълни описаният алгоритъм в кода на героя *podium*, стойностите на променливите **Число1**, **Число2** и **Число3** вече са подредени: **Число1 < Число2 < Число3**.





Как да класирате (сортирате) героите по броя на точките им в състезанието?



Забележка.

В koga е използван блока отиgu go слой от менюто .

Речник

Bubble sort – nogpeжgaне no метода на "мехурчето" За да се анимират героите с изкачване на почетната стълбичка, в кода на всеки робот се прави проверка на кое място е: 1. ако точките на робота са равни на **Число3**, той е първи; 2. ако точките на робота са равни на **Число2**, той е втори; 3. остава възможността роботът да е трети.

За да се осигури последователно придвижване на трите робота, е зададено време за изчакване след проверката и заемането на първо и второ място на почетната стълбичка.



Въпроси и задачи

1. Дадени са числата 21, 39 и 27. Използвайте лист и химикалка и приложете алгоритъма за намиране на най-голямото от трите числа с размяна.

2. За числата 21, 39 и 27 приложете метода на "мехурчето", за да ги подредите. Използвайте хартия и химикалка.

3. Какво трябва да промените в кода на задача 1, за да намерите най-малкото от трите числа? Променете го.

4. Изпълнете задача 2 от урока, като използвате други герои. Задайте имената на героите и променливите на латиница.

5. Друг алгоритъм за намиране на най-голямото от три числа е с използването на допълнителна променлива, в която временно се съхранява поредното най-голямо число. Алгоритъмът е следният: Нека са дадени променливите *A*, *B* и *C*, чиито стойности са числа, а променливата *D* е помощна.

1. Въвеждат се стойностите на А, В и С.

2. На променливата *D* се дава стойността на *A* (първото число).

3. Стойността на D се сравнява с числото B (второто число). Ако B > D, на D се дава стойността на

В и се преминава към стъпка 4, в противен случай се продължава към стъпка 4.

4. Стойността на D се сравнява с числото C (третото число). Ако C > D, на D се дава стойността на C.

5. Променливата *D* вече съдържа най-голямото от трите числа.



Използвайте този алгоритъм и променете кода на задача 1.

Създаване на образователен проект със средствата на изучаван език за блоково



уроци 😸 🗕

Да припомним!

За да се разменят стойностите на две променливи, се използва трета помощна променлива. За да се намери най-голямото от три числа, може да се приложи алгоритъм с размяна или с използване на помощна променлива, в която текущо да се съхранява най-голямото число. Алгоритъмът метод на "мехурчето" се използва за подреждане на три числа. Нека са дадени променливите А, В и С, чиито стойности са числа.

1. Сравняват се A и B. Ако A > B, се разменят стойностите на А и В.

2. Сравняват се *B* и *C*. Ако B > C, се разменят стойностите на В и С.

3. Сравняват се A и B. Ако A > B, се разменят стойностите на А и В.

- **4.** Числата се подреждат: A < B < C.

1. Разгледайте кода на фиг. 1. Кой алгоритъм е представен с него?

Фиг. 1

	число2

2. Дадени са числата 7, 4 и 12. В тетрадките си изпълнете алгоритъма за намиране на най-голямото от трите числа с размяна на стойности.

- Ани създала кода от фиг. 2.
- а) Кои променливи е използвала?
- 6) Обяснете какво е предназначението на кода.

6) Какъв ще бъде резултатът от изпълнението на кода, ако първоначално **число1 = 5**, **число2 = 3** и **число3 = 4**?

Теми за проекти

Изберете си тема за проект от предложените в задачи 4 – 17, върху който ще работите през следващите два часа.

При работата по групов проект по компютърно моделиране трябва:

• да обсъдите темата;

• да определите какво ще създавате – игра, тест, разказ на история, комикс или др.;

• да обсъдите и да опишете дейностите по проекта: определяне на героите и костюмите им и декорите на сцената; описание на сценария; определяне на променливите и как те ще се променят в кода; уточняване на командите и собствените блокове; определяне на необходимите алгоритми; изработване на схема за синхронизиране на действията на героите;

- да разпределите задачите;
- да изпълните задачите;
- да представите проекта пред други хора.

Въпроси и задачи



Забележка

ловете в nankama **uroci 8_9**.

Забележка

Ha cmp. 21 е gageнa paзpaбomka на npoekma – **примерен проеkm 1**.

Упътване

За музикално оформление използвайте файла **BlueDanube**. Добавете го като звук към сцената. Стартирайте го при щракване върху знамето.



Упътване

За определяне на позициите на върховете на правоъгълния триъгълник използвайте квадратна мрежа.

Важно!

1. Създайте подходящ дизайн на вашия проект.

 Задайте начални стойности на променливите.

 Задайте начални позиции на героите, особено на тези, с koumo чертаете.

4. Задайте ограничения на въвежданите от клавиатурата стойности така, че при чертането героят да не допира ръбовете на сцената.

5. Даите указания на потребителя за използване на проекта. **4.** Създайте проект за изчертаване на всяка от дадените фигури и намиране на общата дължина на страните ѝ.



5. Създайте проект "Валс на леда", в който:
а) един герой да изчертава една от фигурите;





б) двама герои едновременно да изчертават фигурите;

6) един или двама герои да чертаят фигури по ваш избор.

6. Създайте проект "Моята България". Използвайте песента "Моя страна, моя България" за озвучаване на анимацията а за изчертаване на националния файл – герой – хвърчило.



7. Създайте проект за изчертаване на квадрат, правоъгълен триъгълник, правоъгълник и успоредник. Потребителят избира вида на фигурата и задава стойности за елементите, необходими за изчертаването ѝ.

8. Отворете файла **4bugs**. Създайте код за четирите буболечки (bugs) така, че те едновременно да описват пясъчен часовник (фиг. 4), като тръгват от центъра на сцената и се движат в различни посоки (фиг. 3). При движението си сменят костюмите си на всеки 10 стъпки.





9. Създайте проект "Калейдоскоп". Четиримата герои – топки, се движат едновременно със стрелки: при натискане на стрелка нагоре се придвижват напред, при натискане на лява или дясна стрелка се завъртат на 15° така, че да очертават едновременно едни и същи фигури (фиг. 5).



Фиг. 5

10. Създайте проект, в който водещият герой:

a) да приканва потребителя да подреди музикалните инструменти по групи според вида им. При подреждането се броят инструментите от всеки вид;

6) да посочва групата с най-много инструменти след приключване на подреждането.

11. Отворете файла **objects**. Създайте код, с който:

a) потребителят да поставя всяка табелка в съответната кутия и да се брои колко табелки има във всяка кутия;

6) да се извежда съобщение за кутията с най-малко поставени табелки.

12. В магазин обявили разпродажба на стоки. Намалението е от 10 до 50%. Създайте проект "Разпродажба", в който на "щанд" в магазина да са показани 3 различни стоки, тяхната цена без намаление и процента на намалението. Изчисляват се цените на стоките след намалението и стоките се подреждат от



ляво надясно по възходящ ред на новите цени.

13. Създайте проект, в който:

a) на сцената за определено време (например 1 или 2 мин) да се показват по случаен начин животни – птици, бозайници и влечуги. Играчът трябва да посочи какъв вид е животното, като въведе от клавиатурата "П" за птици, "Б" за бозайник и "В" за влечуго. Броят се верните отговори за всяка група животни.

6) крайният резултат да се извежда на нова сцена и в края на играта да се показва за коя група животни има най-много разпознавания.

14. Създайте игра "Познай числото". Водещият герой си "намисля" число и казва например: *а) Намислих си число. Умножих го с 5 и получих 235. Кое е числото?* или *б) Намислих си число, разделих го с 3 и получих 1/3. Кое число намислих?* Играта започва с 15 точки. Всеки верен отговор на въпрос от а) или б) носи по 1 точка, а за всеки грешен отговор се отнема по 1 точка. След 5 въпроса водещият герой съобщава общия брой точки, броя на верните отговори за всеки вид въпроси и посочва вида въпроси, за който са дадени най-малко верни отговори. Може да каже и "Двата вида въпроси са с равен брой верни отговори".

15. Създайте игра викторина с въпроси по един от предметите, които изучавате.

16. Създайте игра викторина, с която да проверите знанията на вашите съученици за мерните единици за:
а) дължина; б) повърхнина; в) обем.

17. Съдайте проект "Шевици", подобни на дадените вдясно.



Забележка

На стр. 26 е дадена разработка на проекта – **примерен проект 2**.



Упътване

Помислете каква стратегия може да има водещият герой така, че всеки път да се показват различни числа за въпросите от вида а) и б).



P

урок







Представяне на проект

Задача. Представете проекта си пред класа.

- а) Покажете как се използва.
- **б)** Разкажете каква е целта на проекта.
- 6) Обяснете какви герои и декори сте избрали:
- от библиотеката с вградени герои и декори;
- създадени от вас;
- вмъкнати от готови файлове.

г) Опишете какъв е сценарият на проекта ви и кои са отделните му части.

g) Опишете какви блокове и алгоритми сте използвали:

- от кои менюта;
- какви променливи сте създали и как ги променяте
- какви собствени блокове сте създали и защо сте ги създали;
- какви алгоритми прилагате.

е) Покажете как синхронизирате действията на героите:

- с изчакване;
- с предаване на съобщения.

ж) Разкажете какво участие е взел всеки член на екипа ви в работата по проекта.

След представянето на всеки проект направете оценка на проектите на съучениците си, като използвате критериите, дадени в таблицата.

	Проеkm 1	Проekm 2	Проekm 3	Проekm 4	Проekm 5
1. Героите са пододящо подбрани.					
2. Декорите са пододящи.					
3. Спазен е правописът.					
4. Използвани са правилни блокове.					
5. Проектът е лесен за използване.					
6. Дадени са указания за използване на проекта.					
7. Темата на проекта е интересна.					
8. Проектът е представен добре.					

Примерни проекти (решения)

Примерен проект 1

Създайте проект за изчертаване на всяка от дадените фигури и намиране на общата дължина на страните ѝ.



Решение 1

Екип 1 решил да направи три отделни проекта. За изчертаването на различните фигури избрал различни герои.

Квадрати

За изчертаване на квадратите и намиране на сбора от периметрите им екипът избрал стрелка, която е скрита в настройките на героя. Създал собствени блокове за проверка на въведената дължина на страната на квадрат и за изчертаването му.





Триъгълници

За изчертаването на фигурата, съставена от триъгълници, екипът избрал герой стрелка, скрита от сцената. Създал собствени блокове за проверка на входните данни и за изчертаване на равностранен триъгълник.



Тъй като блокът за проверка на входните данни е един и същ за всички фигури, екипът решил да използва "раницата" в онлайн средата за блоково програмиране и да копира създадения блок в кода на останалите проекти.







Забележка

Решенията на ekun 1 може ga виgume във файловете squares, triangles, carousel u flower.

Въртележка

Екипът избрал котето за основен герой и го скрил от сцената. Установил, че всяка перка от въртележката всъщност е половината от ромб. Освен това екипът решил да остави на потребителя да прецени на какъв ъгъл да се завърта "перката", за да се получи въртележката. Създал собствени блокове за проверка на входните данни и за изчертаване на "перката" по дадено рамо и остър ъгъл. Изчертаването на всички рамена на въртележката е описано в цикъл, който се повтаря *360/големината на острия ъгъл на ромба* пъти.



Цвете

Екипът решил да прояви творчество и да добави още една фигура – цвете, образувано от ромбове, като използва вече създадения код за изчертаване на въртележката.

В края на основния код добавил блок за показване на котето и създал код, с който, щом се щракне върху котето, то се скрива, а цветето започва да се върти.





Решение 2

Екип 2 решил да въведе водещ герой, който да дава информация на потребителя за фигурите и резултатите от намирането на общата дължина на страните им, като използва герой под формата на топка за изчертаване на фигурите. Установил, че всяка фигура се състои от съставяща я фигура – съответно от равностранен триъгълник, , квадрат и перка, която е половината от ромб с остър ъгъл, равен на *360/броя на перките във въртележката*. Екипът решил водещият герой да изисква да се въведе дължината на страната на съставящата фигура, като се прави проверка дали въведената дължина е по-малка от 90 стъпки.

Въртележката и фигурата от триъгълници се разполагат в центъра на сцената. Фигурата с квадратите се разполага така, че долният ѝ ляв ъгъл да е в позиция (-240; -170).

Всяка фигура се изчертава на отделна сцена. Използват се собствени блокове за изчертаване на съставящата я фигура и изчисляване на периметъра ѝ.

Екипът предвидил разстоянието между квадратите да бъде 30 стъпки; участниците в него преценили, че за да се виждат всички квадрати на сцената, дължината на страната на квадрата не трябва да бъде по-голяма от 90.













Забележка

Moжете да видите рзултата от проекта във файла **figures**. Фигура от квадрати







Код на водещия герой

когато 💻 е цракнато	
И изтрий всичко	
направи периметър – на 0	
промени костюм на abby-а 🕶	
кажи Здравей, сега ще изчертаем няколко фигури, съставени от изучавани в 5. клас геометрични фигури. за (5)	ек
следващ костюм	
питай Въведи дължина на страната на съставната фигура, не по-голяма от 90 стъпки. и чакай	
спедващ костюм	
повтаряй докато отговор < 91	
питай Въведи дължина на страната на съставната фигура, не по-голяма от 90 стъпки. и чакай	
следващ костюм	
разпространи триъгълници – и чакай	
кажи Тази фигура се състои от три равностранни триъгълника със страна за 4 сек	
следващ костюм	
кажи оттовор за 2 сек	
следващ костюм	
кажи съедини Периметърът ѝ е равен на периметър за 2 сек	
следващ костюм	
разпространи квадрати 👻 и чакай	
кажи Тази фигура се състои от четитри квадрата със страна за 4 сек	
следващ костюм	
кажи оттовор за 2 сек	
следващ костюм	
кажи съедини Периметърът й е равен на периметър за 2 сек	
разпространи въртележка 👻 и чакай	
следващ костюм	
кажи Тази фигура се състои от осем половинки на ромб със страна за 4 сек	
следващ костюм	
кажи оттовор за 2 сек	
следващ костюм	
кажи съедини Общата дължина на перките е равна на периметър за 2 сек	
следващ костюм	



Kakвu герои и gekopu ga се изберат?



Какъв е сценарият на проекта?



Какви алгоритми ще се приложат в проекта?

Какви променливи са необходими за проекта?

Важно!

Имената на променливите трябва да подсказват тяхното предназначение.

Какви блокове са необходими?

Примерен проект 2

В магазин обявили разпродажба на стоки. Намалението е от 10 до 50%. Създайте проект "Разпродажба", в който на "щанд" в магазина да са показани 3 различни стоки, тяхната цена без намаление и процента на намалението. Изчисляват се цените на стоките след намалението и стоките се подреждат от ляво надясно по възходящ ред на новите цени.

Екипът, разработил проекта, решил стоките в магазина да са дрехи. Избрал ги от от библиотеката с герои на **Scratch** и



променил имената им на български език. За подреждане на стоките по цена след намалението е избран герой – бутон, от библиотеката. Към костюмите на бутона са добавени текстовете "Подреди" и "Подредено". Декорът на сцената е създаден във векторен режим на графичния редактор на средата. Поставени са надписи "Найниска цена" и "Най-висока цена".

При започване на проекта героите са поставени на различни позиции – панталон: (-160; 40); тениска: (0; 40); рокля: (160; 40). Показани са цената на всяка стока и намалението ѝ в проценти. Щом се щракне с мишката върху стока, цената ѝ се намалява със съответния процент и на сцената се показва новата цена. Цените и процентите за намаление са зададени в кода на проекта. Когато се щракне върху бутона **Подреди**, стоките се подреждат според намалената им цена и бутонът за подреждане сменя костюма си на **Подреден**.

За изчисляване на намалените цени на стоките трябва да се използват формули. Ако стоката има първоначална цена a и намаление b%, новата цена след намалението ще е a-a*b/100. За подреждане на стоките по новата им цена, от най-ниска към най-висока, ще се използва алгоритъм за подреждане на три елемента с разместване (метод на "мехурчето").

За всяка стока са необходими три променливи: за старата ѝ цена, намалението в проценти и новата ѝ цена. Тези променливи трябва да действат само за съответния герой. Например променливите за панталона може да са: **цена_п** – за първоначалната цена, **намаление_п** – за намалението на в проценти и **нова_цена_п** – за намалената цена на панталона. За реализирането на алгоритъма за подреждане ще се използват променливи **цена1**, **цена2**, **цена3** и **размяна**, които са създадени за всички блокове.

Необходими са блок отиди до за задаване на начални позиции на стоките, блокове за задаване на начални стойности на променливите и за скриване на променливата за новата цена на всяка стока.



При начално стартиране в кода на бутона **Подреди** трябва да се зададе неговата начална позиция и началните стойности на променливите **цена1**, **цена2** и **цена3**.

За изчисляване на намалената цена ще се използват собствени блокове към всеки от героите.



Тук се използва променливата **намалена_цена**, която е създадена за всички герои. Чрез нея се предава стойността на новата цена към съответните променливи – **нова_цена_п** и **цена1**, когато блокът е в кода на героя – панталон, **нова_цена_т** и **цена2**, когато блокът е в кода на героя – тениска, и **нова_цена_р** и **цена3** за героя – рокля. При щракване върху всеки герой – стока, се прави пресмятане на новата цена със собствения блок **намали ... с проценти ...** В кода на героя – панталон, на променливите **цена1** и **нова_цена_п** се дава стойността на променливата **намалена_цена**. В кода на ге-

роя – тениска, на променливите цена2 и нова_цена_т се дава стойността на намалена_цена. В кода на героя – рокля, на цена3 и нова_цена_р се дава стойността на намалена_цена.

Подреждането на променливите цена1, цена2 и цена3 се осъществява в кода на бутона за подреждане с алгоритъма за размяна на две съседни стойности. В резултат на този алгоритъм стойностите на променливите цена1, цена2 и цена3 са подредени:



цена1 < цена2 < цена3.

Използва се съобщение "Подреди" към героите – стоки, за направеното подреждане на цените, в резултат на което те трябва да заемат съответното място. След подреждането бутонът сменя костюма си с този с надпис "Подредено".

За да се определи кой герой на коя позиция трябва да отиде, неговата цена се сравнява с променливите цена1, цена2 и цена3: ако съот-

ветната променлива **нова_цена_**... е равна на **цена1** (най-ниската), героят се позиционира вляво, ако е равна на **цена2**, се позиционира в средата, а ако е равна на **цена3** (най-високата), се позиционира вдясно. Това става тогава, когато всеки герой получи съобщението "Подреди".





Упътване

Може да направите собствен блок към един от героите– стока и след това да го копирате в кодовете на другите герои.



Забележка

Можете да видите крайния резултат от проекта във файла **sale**.

Издателска група

ПРОСВЕТА е член на Асоциацията на европейските издатели на учебници EUROPEAN EDUCATIONAL PUBLISHERS GROUP

www.prosveta.bg www.e-uchebnik.bg www.e-prosveta.bg